March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

Viterbi Based Document Classification with Long Short Term Memory

[1] Vetriselvi T, *Dept of Computer Science and Engineering, K.Ramakrishnan College of Technology, Tiruchirappalli, Tamilnadu, India.*
[2]Gopalan NP, *Dept.Of Computer Application, National Institute of Technology, Tiruchirappalli, Tamilnadu, India.*
[3] Kairunisha Bee K, *Dept of Computer Science and Engineering, K.Ramakrishnan College of Technology, Tiruchirappalli, Tamilnadu, India.*
Email: vetriselvi09@gmail.com

Article Info
Volume 83
Page Number: 6042 - 6050
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 01 April 2020*

Abstract:

Nowadays, the number of documents published are widely increasing, for the efficient retrieval of the document, classification of the document based on the similarity to the particular domain needs to be done and then the ranking of papers also need to be done, which results in the effective retrieval of the document. This document classification is very helpful in the information retrieval system, which reduces the time of retrieval. There are different methods for classification, like Naive Bayes Classifier, Support Vector Machines, Artificial Neural Network, Latent Semantic Indexing, K-Nearest Neighbor Algorithm, etc. In this paper we use Latent Semantic Indexing (LSI)is one of the best approaches for classification of the document which uses the mathematical method called Singular Value Decomposition(SVD) for classification of documents. The Parts of Speech (POS) tagging are also used in these types of classification which helps to improve the performance of the system. We use the Viterbi algorithm along with Long Short Term Memory(LSTM) to find the POS tags for each word in the document. The LSTM and bidirectional LSTM is also compared by combining different POS tag identification algorithm and the best method is used in this architecture of theproposed system of classification.

Keywords: POS Tagging; Latent Semantic Indexing(LSI); Cosine Similarity; Singular Value Decomposition; Long Short Term Memory (LSTM);Bidirectional LSTM(Bi-LSTM)

I.INTRODUCTION

With the rapid growth technologies, researches on different domains result in many new innovative ideas, which is the base for the generation of the number of documents. As many documents are generated, those documents need to be classified based on their domain which will be helpful for efficient retrieval of the document within a short period. Classification is nothing but the categorization of those content classes of predetermined categories.There are different methods for the classification of the document based on the required domain. Latent semantic indexing (LSI) is a method that aims to identify the statistical association of terms. LSI uses a statistical technique called Singular Value Decomposition (SVD). This is a technique that identifies the relationship between the terms and concepts in an unstructured collection of text. This technique is used to decompose the matrix. Mostly the preprocessing is done by stemming and stop word elimination; in our proposed system we first identify the POS tags of every word in the sentence. Here we analyze different approaches for POS tag identification and the best approach with greater efficiency is selected. The following approaches which are analyzed and compared in the POS tag identification are Hidden Markov Model(HMM), Viterbi Algorithm, LSTM along with HMM, LSTM along with Viterbi, bidirectional LSTM with HMM and Bidirectional LSTM with Viterbi. Then only the words with

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

useful tags are taken into the account for construction of the TF-IDF matrix. Then, the cosine similarity is used to find the value of relationships between the document and the domain, with those values ranking is done.

The remaining part of the paper has organized as follows: in section II, we describe the related work along with theexplanationsabout POS Tagging, Hidden Markov Model, Viterbi Algorithm. In section III we convey the proposed system by explaining the two phases of the architecture. The model analyzedand evaluated in Section VI. It is concluded in Section V along with future scope.

II. Related Works

To retrieve effective the document effectively, there is a need for classification of document to the domain in which they come under. We use latent semantic indexing (LSI) along with cosine similarity in this paper.

In the traditional system of classification mostly vector space model is used, in [2] LSI is used for Arabic text classification. SVD is used to extract the text from 400 documents that are used for testing. This paper also shows cosine similarity is the best technique than the Support Vector Machine(SVM).

To assign most appropriate labels to a specific document and it also compares Naïve Bayes and Random Forest with various document representation techniques such as TF-IDF, LDA, DOC2VEC. In which we take TF-IDF for our proposed system[3][16].

For the retrieval of the document in which the prepossessing steps such as Tokenization Stop word elimination, stemming, dictionary format of the word, etc are used which are the time-consuming process[5]. This concept of preprocessing also used for the bengala document ranking system along with LSI which gives an accuracy of 88.557% [13]. These papers also explain LSI along with SVD, which overcomes the vocabulary problem and also helpful in finding textual information.[5][9][13][12]. Latent semantic analysis (LSA) is also used to get useful information and also for the classification of big data. LSA also reduces the time complexity which helps in the classification of big data which also uses cosine similarity measure [11]. The PNP files classification with latent semantic indexing technique is shown in [12].

In some paper, the graph-based approaches are discussed which gives more accuracy, but it will need more dimensional space and it will be very difficult to construct a graph for all the words in the document. It will be a confusing task and it will be difficult to understand and implement [15][17].

For finding the similarity between the document and the query in the information retrieval system, different similarity measures like jaccard similarity; cosine similarity is discussed in which we use cosine similarity in our proposed model.

As these papers use stemming, stop word elimination for prepossessing this filtration process the number of word for TF-IDF get reduces. This kind of approach will make the system to struck when the search keywords are not matched with the documents, word sequencing problem will arise, so simplification needs to be done. We use the POS tagging approach instead of this preprocessing where the tags are divided into useful and useless tags and only the useful tags are taken for further processing. For the identification of POS tags we use the Hidden Markov Model and The Viterbi Algorithm, these two algorithms are compared for identification of POS tags for Nepali text. Where the Viterbi algorithm is found to be a faster HMM model with an accuracy of 95.43%[6]. Similarly in POS tag identification of Gujarati text Viterbi algorithm performs well than the HMM with the accuracy of 92.8%[7].when we use Viterbi and SVM for Gujarati POS Tag identification, Viterbi algorithm performs well when compared with SVM in the examination of 1700 words from Gujarati corpus[8].

The performance can we enhanced when the LSTM is used along with POS tagging. Itis used for sentence representation. In which tagging is used for

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

static representation and memory storing is used for structural representation than these two properties are combined in POS LSTM. This model gives higher quality sentence representation than the traditional model by capturing better syntactic information [1]. When POS tagging is combined with the advantage of LSTM which is used to find the tags more accurately which uses temporal sequences and their long-range accuracy more effectively this LSTM and the bidirectional LSTM are compared and the best approach is used in the proposed system.

2.1Parts Of Speech Tagging

Parts of speech tagging is identifying the parts of speech for each word in a sentence. In other words, it is the identification of grammatical category of each word in a sentence comes under. The word can be classified as noun, verb, determiner, preposition, pronoun, adjective and adverb. Mostly POS tagging is used which consists only of a few sets of POS tags which may also not used to give more details about the grammatical categories. For more fine details about the word " penn tree bank tag set " is used which contains 48 tags which give a clear understanding of the grammatical category under which the word present in the sentence. A single word can come under different POS tags. To identify the exact part of the speech tag, the entire sentence has to be analyzed.

Experiment: considering the word "success"

1. The new tv show has been a big success.
2. You need to work hard if you need to succeed.

Here, in this first sentence, the word success is in the form of a noun but it is the second sentence it is in the form of verb.

Consider a corpus the word success noun 95% of the time and as a verb 5% of the time. To give the exact tag to each word generative model and discriminative or conditional model are used the vg - starting, nm/noun starting, reactive and convert methods to find the POS tags based on the grammatical constructions of basic lexical phrases, this may lead to creating a problem as if we categorize all these sentences under these three criteria. In the generative model, data are generated based on the class p (d | c) example, Hidden Markov Model, Naive Bayes Classification, etc.The logistic regression, maximum entropy model are conditional models used with probability p(C/D)

2.2. Hidden Markov Model

Hidden Markov Model is a generative model used for assigning a most suitable tag for each word in a sentence that uses probability for finding the exact tag .the tag with maximum probability is chosen for that particular word in the corpus. The 4-gram HMM is used in which two probabilities are used transition and emission probability, which may consume more time as it has several comparisons, for efficient computation we use the formula,

$\hat{T}$ = $argmax_T \prod P(w_i|w_1 \dots w_{i-1}, t_1 \dots t_i)P(t_i|t_1 \dots t_{i-1})$ (1)

Formula derivation:

Let w represents the words where w = $w_1 \dots w_n$ and t represents the tags where t = $t_1 \dots t_n$

$\hat{T} = argmax_T P(T|W)$

$= argmax_T \frac{P(W|T)P(T)}{P(w)}$

$= argmax_T P(W|T)P(T)$

$= argmax_T \prod P(w_i|w_1 \dots w_{i-1}, t_1 \dots t_i)P(t_i|t_1 \dots t_{i-1})$

The probability of a word depends only on its ownPOStag: $P(w_i|w_1 \dots w_{i-1}, t_1 \dots t_i) = P(w_i|t_i)$

Bigram assumption: $P(t_i|t_1 \dots t_{i-1}) = P(t_i|t_{i-1})$

Using these simplifications: $\hat{T} = argmax_T P(w_i|t_i)P(t_i|t_{i-1})$ (2)

WordLikelihood Probabilities: $P(w_i|t_i)P(t_i|t_{i-1})$ (3)

Tag Transition Probabilities: $(t_i|t_{i-1}) = \frac{c(t_{i-1},t_i)}{c(t_{i-1})}$ (4)

Algorithm 2.1 : Hidden Markov Model algorithm for POS tagging

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

Input: Corpus for which POS tag need to be identified
Output: Words correspondingPOS tags
Step 1:Consider a corpus for which we have to find the parts of a speech tag.
Step 2: Consider all the possible tags and form a set. Use the likelihood probability formula (3) and find the probability of the word and all its possible tags.
Step 3: The probability of each word and the possible tags are stored in a table.
Step 4: Likewise for all the words in the corpus the steps 2 and 3 are done continuously till the end of the corpus.
Step 5: Then the tag with the highest probability is chosen as the exact matching tag.
Let us consider the working of HMM,we consider a sentence "promised to write a poem" for which we are going to find the POS tags. For each word, we consider all the possiblePOS tags with all possible flow which is shown in Fig.1 Then the likelihood probability of each word is calculated for all possible tags by computing the path value.

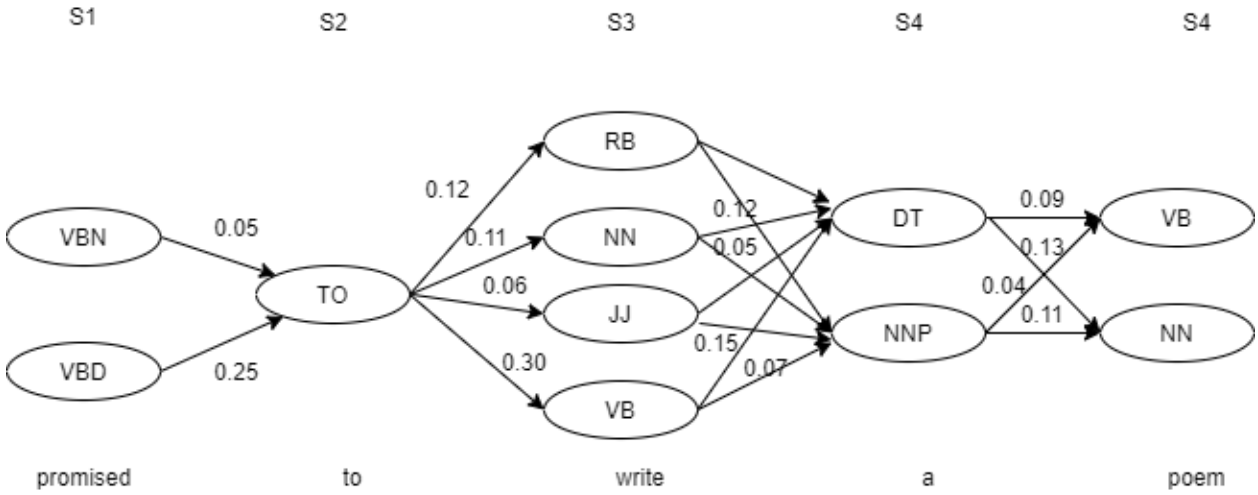

Fig.1. Example for Hidden Markov model

To find the best tag sequence we need to consider all the possible paths and we need to calculate the probability using formula (1) or with formula (2) which is the simplified formula to calculate the probability, the path with the higher probabilistic value is considered as the best tag sequence. Here we consider six possible paths, for which the path probabilistic calculation is done. Table1 shows the paths with their probabilistic value. The tag transition probability can be calculated using formula (4).

S.No	Different paths	Probabilistic value
1	VBN-TO-NN-DT-VB	0.37
2	VBN-TO-VB-DT-NN	0.63
3	VBD-TO-VB-DT-NN	0.83
4	VBD-TO-VB-NNP-VB	0.66
5	VBD-TO-NN-NNP-VB	0.45
6	VBN-TO-NN-DT-NN	0.41

Table1.Possible paths and their probabilistic

From Table.1, we can find that the path "VBD-TO-VB-DT-NN" has the highest probability of 0.83, compared to other possible paths. So we consider "VBD-TO-VB-DT-NN" as the best path. The constructed path is shown in Fig.2.

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

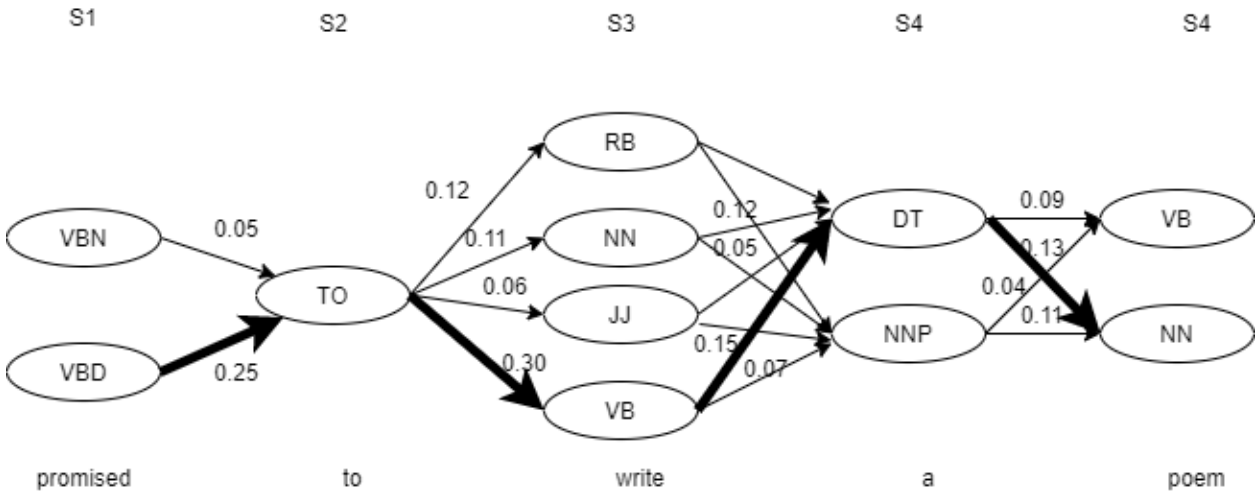

Fig.2.
Best Tag sequence for the sentence.

2.3 Viterbi decoding algorithm

In theHMMmodel finding the probability of each word with all the Possible tags is very difficult and it's a time-consuming process so the Viterbi algorithm is used. Viterbi algorithm finds the best path of tag sequences under which a sentence is formed. The best path is computed by considering all the PossiblePOS tags for each word and then considering the most appropriate later computation tag and then backtracking is used to find the best predecessor tag. This Viterbi algorithm is a dynamic programming approach that is used to minimize the computational time. To find the best path we need to consider the cheapest cost of the state and then backtracking is used.

Algorithm 2.2 : Viterbi Algorithm For POS tagging

Input: Sentence for which POS tag need to be identified

Output: Words corresponding POS tags

Step1: Consider a sentence for which we need to find the best path of POS tagging.

Step2: Consider a set for each word which consists of all the PossiblePOS tagging for that particular word.

Step3: Consider the two best path which has more appropriate POS tagging. It may consist of same later computations

Step4: Then the cheapest cost can be computed by using the formulas

$$\delta j(s+1) = max1 \le i \le N\delta i(s)p(t_j|t_i)p(w_{s+1}|t_j)$$

Step5: Then backtrack from that state to best predecessor state is done by using the formula

$$\varphi j(s+1) = argmax1 \le i \le N\delta i(s)p(t_j|t_i)p(w_{s+1}|t_j)$$

Step6: Continue step 4 and step 5 until we find the POS tag for all the words in the sentence.

III. Proposed System

The aim of the proposed system is to classify the documents based on their domain. The main architecture is divided into two phases the first phase is the preprocessing phase and the second is the final processing phase.

A. Preprocessing phase

In this preprocessing phase, the architecture consists of five main parts. They are POS tag identifier with LSTM, Tag extractor, Lemmatization, calculating term weigh, and then matrix construction. Fig.A, shows the preprocessing phase architecture.

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

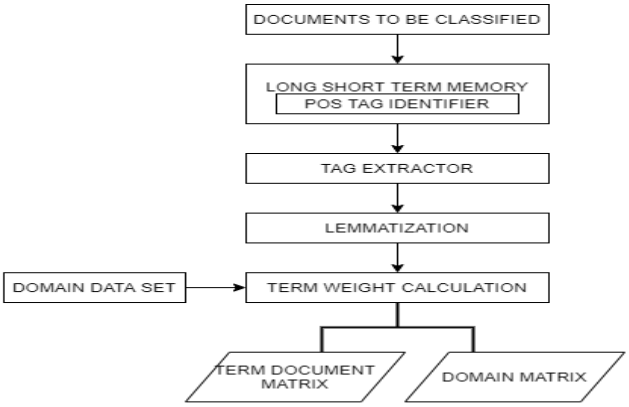

Fig.A
Architecture Of Pre Processing Phase

1)POS Tag Identifier and LSTM:

Set of documents are considered for classification. For the document each sentence is separated,and then the POS tag for each word in the sentence is identified using the Viterbi decoding algorithm. In section II, the POS tagging identification algorithm and the formulas used in those algorithms are explained. The HMM model and the Viterbi model are discussed for calculation of POS tags Algorithm 2.1 shows the steps for POS tag identification using HMM and Algorithm 2.2 shows the step for POS tag identification using Viterbi. Then the LSTM[1] is used as a buffer to store those POS tags of the sentence and compute the tags.It has the quality of maintaining the information for a long time as it consists of memory cells. This will be help full in finding the exact POS tag for the sentence as the previous computation is stored in the memory as a loop. This LSTM is one of the important factor which plays a main role to improves the performance of the system.

Is also compared when bidirectional LSTM, the bidirectional LSTM is used to run the input in two ways, one from past to future and one from future to past. But the unidirectional LSTM runs only backward. This Bi-LSTM increases the performance by preserving the information from both past and future.The two way approach performs well as they can understand the context better.

2) Tag Extractor:In this modulePOStags which are identified in the previous step are categorized into two categories useful and useless tags. Useful tags are the verb, noun, adjective, and adverb. Other tags are useless tags. The words with useful tags are taken for the next step other tags are removed.

3)Lemmatization:

The words with useful tags enter this module. Lemmatization is a methodology used to find the dictionary for the word by removing the inflectional ending of the word. Now the root word is generated which is called a lemma, this root form of the word will be more helpful in the construction of the term context matrix.

4) Term Weighting:

The words after lemmatization goes into this process there the number of time the word occurs in the document is calculated and this is done for all the words in which are filtered after lemmatization. This term weighting is also done for the word in the data set in that particular domain.

It creates two matrixes, one for the document and another related to that particular domain. This is the output of this first phase.

B)Final Processing Phase:

This final processing phase plays a main role where the classification of the document takes place. For document classification, we use Latent Semantic Indexing and Cosine Similarity. Then the ranking of the document is done. Fig.B, shows the architecture diagram of the final processing phase.

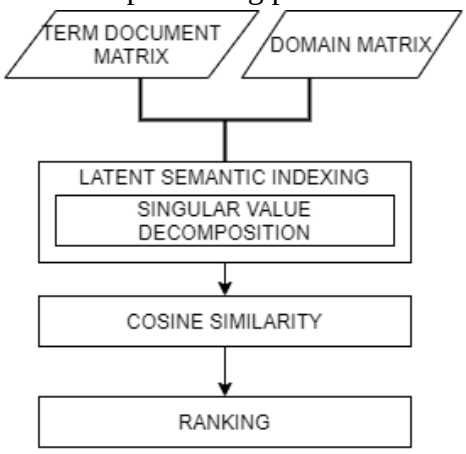

Fig.B. Final Processing Phase Architecture

1) Latent Semantic Indexing:

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

After the preprocessing step we get two matrixes these matrixes document matrix A and the domain matrix D. These are the input to LSI. LSI is a technique in natural language processing it is used for identification of the relationship between the document and their domain[5]. This LSI uses a mathematical technique called singular value decomposition (SVD). This technique is used to reduce the number of rows and used to calculate the similarity. The SVD decomposes the term-document matrix into three matrices U, V, S where U is left singular matrix, V is right singular matrix, and S is the diagonal matrix.$A = USV^T$ is the matrix decomposition formula.

Where, U^TU and V^TV are the identical matrix. For svd we need to calculate the Eigenvalues and eigenvectors of AA^T and A^TA. The ortho normal eigenvectors of AA^T is the column of Vand the ortho normal eigenvectors of A^TA is used to calculate U. The singular values in s are calculated by the square roots of Eigenvalues of AA^TorA^TAin descending order. $W\vec{x} = \lambda\vec{x}$is a formula which is used to calculate the Eigen value and the Eigen vector. Where, W is a square matrix from the term document matrix, $\vec{x}$ is a nonzero eigenvector of W and λ represents a scalar value. Here, λ is said to be an Eigenvalue of a and $\vec{x}$is an eigenvector of a associated to λ. The λ value is used to determine the values of three matrices - U, S and V. Now we deduct the irrelevant or less important topic which has a rare contribution in the document scoring. Since in the S matrix, elements are organized in descending order, the lower values indicate that corresponding topics are less important or sometimes they are irrelevant. For these reasons, by neglecting the lower values, we reduce the dimension about 5% from the total dimension (n x n), then we get a new dimension i.e. K x k for the matrix S. The new column value K is used to all other matrices as$U \approx U_k$, $S \approx S_k$, $V \approx V_k$, and $V^T \approx V_k^T$. Here, rows of $V_k$holds the eigenvalues. These are the coordinates of the individual document vectors. In this way, we get $\overrightarrow{doc1}$, $\overrightarrow{doc2}$, …, $\overrightarrow{docn}$ as document vectors for n documents.The domain vector $\vec{d}$is generated with the D, U and S matrices and matrix operations among them.

$$\vec{d} = D^T U_K S_K^{-1}$$

(5)

2)Cosine similarity:

Cosine similarity is used to find the closeness between two vectors this shows how the document is related to the domain[2]. We use the document vector$\overrightarrow{doc1}$ and the domain vector$\vec{d}$ which is calculated from the formula (5) is used for finding the similarity

$$sim\left(\overrightarrow{doc}, \vec{d}\right) = \frac{\overrightarrow{doc} \cdot \vec{d}}{|\overrightarrow{doc}|\,|\vec{d}|} \qquad (6)$$

In the given formula (6) is used to calculate the similarity value the output of this formula ranges from -1 to +1. The closeness of the document with domain is identified by their similarity value, intuitively the higher similarity shows that they are close and low similar vice versa.

3)Ranking:

From the value generated of the above step the ranking is done. The document which are highly related to the domain are kept first, likewise the ranking is done.

IV.Experiment Results

As we use POS tagging instead of preprocessing steps we find which algorithm gives better result by calculating the accuracy and time taken for various approaches. We consider a document for which various POS tag identification techniques are used to identify POS tags. Table2 describes the different POS tag identification approaches and the accuracy value from those approaches.

Techniques	**Accuracy**
Hidden Markov Model (HMM)	76.97%
Viterbi Algorithm	93.43%
LSTM based HMM	80.40%
LSTMbased Viterbi	95.36%

Published by: The Mattingley Publishing Co., Inc.

March-April 2020
ISSN: 0193-4120 Page No. 6042 - 6050

Bidirectional LSTM with HMM	85.72%
Bidirectional LSTMwith Viterbi	96.61%

Table2. Comparison of different POS tag identification Approaches.

Bi-directional LSTM along Viterbi gives improved accuracy then other methods of tagging. Then the useful tagged words are separated from the document after tagging. Then all the process of our proposed system is carried out with the number of documents and different data set. In this experiment we are going to consider 250 documents, we split into 5 categories each of 50 documents. In our proposed system we use Wikipedia data sets that is Text data and News Articles data set for classification. Then the precision, recall, accuracy are calculated.

A. Precision:

Precision is considered as the calculation of positive predictive value. It is used to calculate the number of documents that are classified correctly to the domain to the total number of documents classified in that particular domain. It is also defined as the percentage of the document that matches correctly to the domain. Precision is measured by using the following formula,

$$P = \frac{a}{a+b} \quad (6)$$

Where, P = precision. a = number of classified documents that matches correctly to the domain.

b = number of classified documents that does not matches correctly to the domain.

B. Recall

Recall is considered as the classification to calculate the sensitivity of the documents. It is used to calculate the fraction of documents that are classified to the particular domain over the total number for documents relevant to that particular domain. On the other hand, recall is the percentage of correct documents that are classified. Recall value can be measured by using the following formula,

$$R = \frac{a}{a+c} \quad (7)$$

Where, R = recall. a = number of correct documents that are classified. c = number of correct documents that are not classified.

C. Accuracy

The Accuracy is calculated using precision and recall.

$$Accuracy = \frac{2*P*R}{P+R} \quad (8)$$

Where P = precision,

R= recall.

Now the experiment with 250 documents is carried out and with formula 6 precision is calculated , with formula (7) recall is calculated and the accuracy is calculated with the help of formula (8) for all the five categories of documents then from the calculated value the total accuracy of the system is calculated. Table3 shows the experimental results.

Categories	**Precision**	**Recall**	**Accuracy**
Category 1	97	96	96.49
Category 2	95	99	96.95
Category 3	96	100	97.95
Category 4	100	98	98.98
Category 5	95	100	97.43
Average	**96.6**	**98.6**	**97.56**

Table3. Experimental result

The above Table3 shows the accuracy of the proposed system (97.56%) which is higher than the accuracy of the other proposed systems.

V.CONCLUSION

In our document classification and ranking system, we have presented a approach which shows the 97.56% accurate result. This accuracy is achieved as the preprocessing steps like Stop WordElimination, stemming, etc are removed and instead of that we use POStag identification alone with the long short term memory. This makes our system more efficient then the other. This papers also compares the tag identification algorithms along with LSTM and bidirectional LSTM. This system can be further

Published by: The Mattingley Publishing Co., Inc.

expanded and used in the medical field for disease prediction.

REFERENCES

[1] Wenhao Zhu, Tengjun Yao, Wu Zhang and Baogang Wei, "Part-of-Speech-Based Long Short-Term Memory Network for Learning Sentence Representations", in IEEE Access.

[2] Fawaz S. Al-Anzi and Dia AbuZeina ," Towards an Enhanced Arabic Text Classification Using Cosine Similarity and Latent Semantic Indexing", Journal of King Saud University - Computer and Information Sciences.

[3] Donghwa Kim, Deokseong Seo, Suhyoun Cho and Pilsung Kang," Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec", Information Sciences.

[4] George W. Furnas, Scott Deerwester, Susan T. Dumais, Thomas K. Landauer, Richard A. Xarshman, Lynn A. Streeter, Karen E. Lochbaum," Information Retrieval using a Singular Value DecomPOSition Model of Latent Semantic Structure", ACM SIGIR Forum august 2017.

[5] Neelam Phadnis and Jayant Gadge," Framework for Document Retrieval using Latent Semantic Indexing" ,International Journal of Computer Applications May 2014.

[6] Archit Yajnik," Part of Speech Tagging Using Statistical Approach for Nepali Text", World Academy of Science, Engineering and Technology International Journal of Cognitive and Language Sciences Vol:11, No:1, 2017.

[7] Archit Yajnik and Manisha Prajapati," Part Of Speech Tagging Using Statistical Approach For Gujrati Text ", International Journal Of Applied Research In Science And Engineering.

[8] Manisha Prajapati and Archit Yajni," POS Tagging Of Gujarati Text Using VITERBI And SVM",International Journal of Computer Applications (0975 – 8887) Volume 181 – No. 43, March 2019.

[9] Andri Mirzal," Clustering and latent semantic indexing aspects of the singular value decomPOSition",International Journal Of Decision Science, Vol 8,No.1,2016.

[10] Abhishek Jain, Aman Jain, Nihal Chauhan, Vikrant Singh and Narina Thakur," Information Retrieval using Cosine and Jaccard Similarity Measures in Vector Space Model", International Journal of Computer Applications (0975 – 8887) Volume 164 – No 6, April 2017.

[11] Hadeel Alazzam and Abdulsalam Alsmady," A Distributed Arabic Text Classification Approach Using Latent Semantic Analysis for Big data",2017 12th International scientific and Technical Conference on Computer Science and Information Technologies.

[12] AngelicaM.Aquino and EnricoP.Chavez," Analysis on the use of Latent Semantic Indexing (LSI) for document classification and retrieval system of PNP files", MATEC Web of Conference 189, 03009(2018).

[13] Md. Nesarul Hoque, Rabiul Islam and Md. Sajidul Karim," Information Retrieval System in Bangla Document Ranking using Latent Semantic Indexing", 1st International Conference on Advances in Science, Engineering and Robotics Technology 2019 (ICASERT 2019).

[14] Yue Liu," Investigation of Viterbi Algorithm Performance on Part-of-Speech Tagger of Natural Language Processing", 2017 International Conference on Computer Systems, Electronics and Control (ICCSEC).

[15] Zhilong Zhen and Juxiao Zhang," Text Classification Based on Latent Semantic Indexing and Graph Embedding", 2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD).

[16] T.Vetriselvi,N.P.Gopalan," An Improved Key Term Weightage Algorithm For Text Summarization Using Local Context Information And Fuzzy Graph Sentence Score", Spinger, March 2020

[17]T.Vetriselvi,N.P.Gopalan," A Novel Approach to Summarization based on Centroid Fuzzy",International Journal of Engineering and Advanced Technology 8 (4c), 86-90